FIG. 1

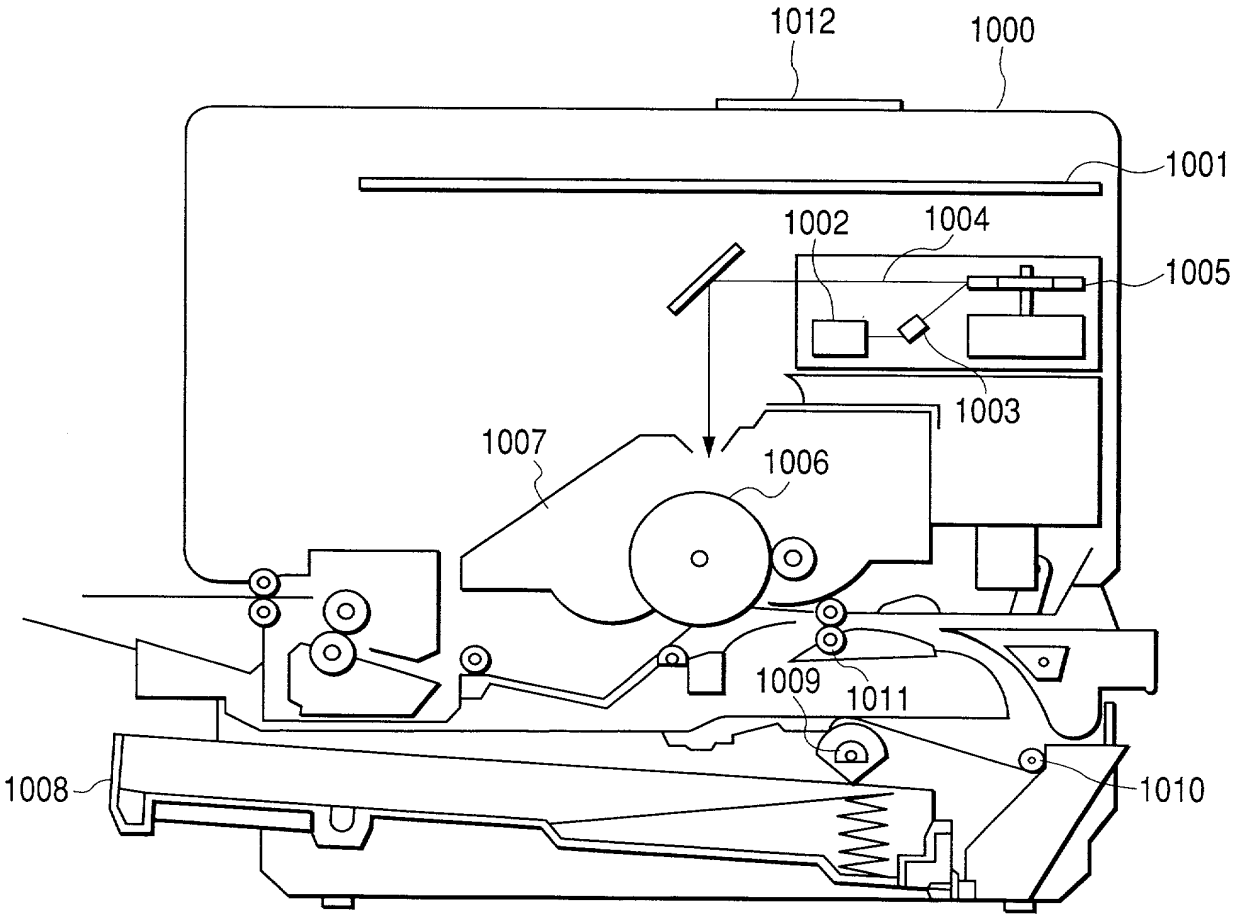

1012
1000
1001
1002
1004
1005
1003
1007
1006
1009
1011
1008
1010

FIG. 2

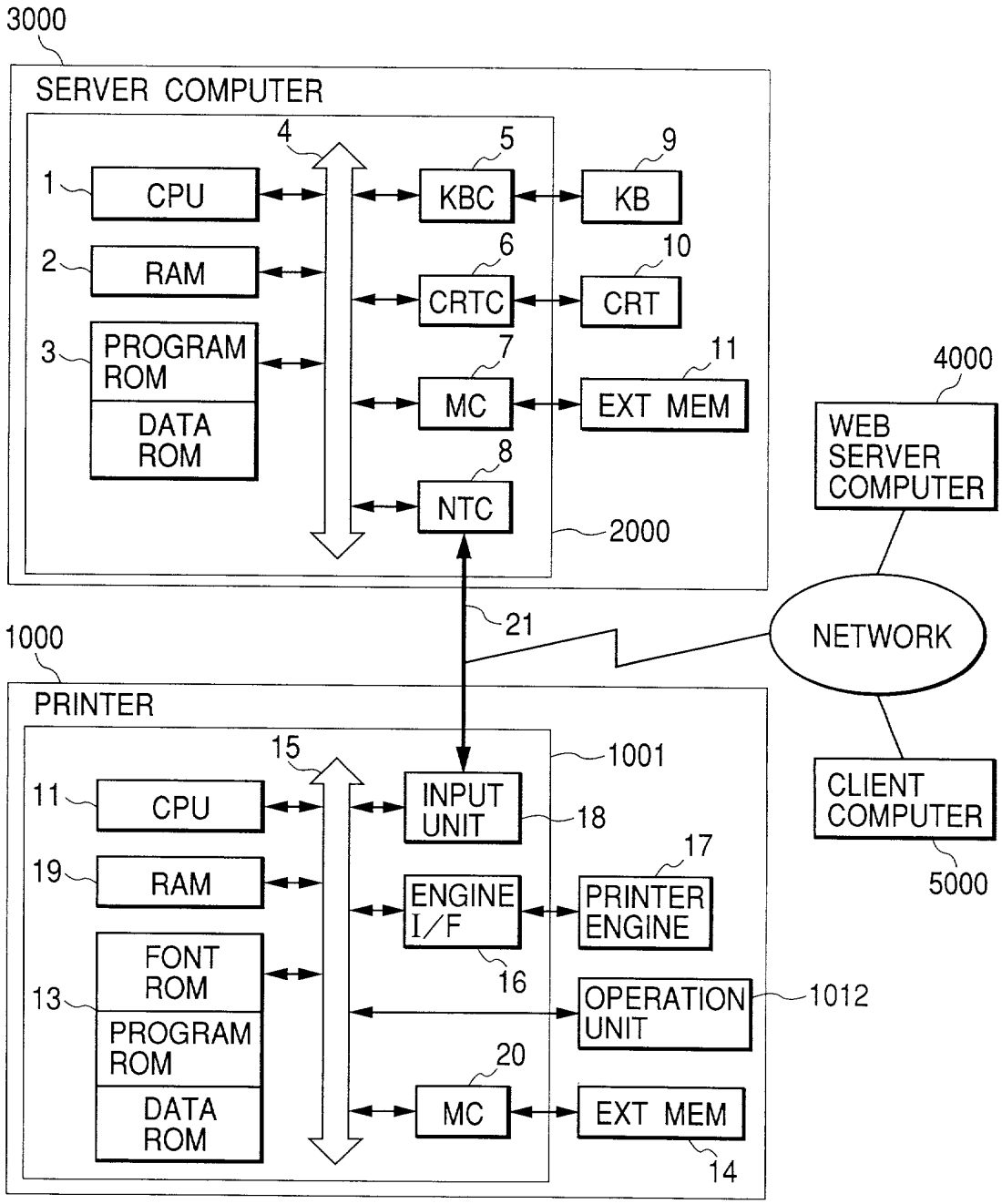

3000
SERVER COMPUTER
4
5
9
1
CPU
KBC
KB
2
RAM
6
10
CRTC
CRT
3
PROGRAM ROM
DATA ROM
7
11
MC
EXT MEM
8
NTC
2000
4000
WEB SERVER COMPUTER
21
NETWORK
1000
PRINTER
15
1001
11
CPU
INPUT UNIT
18
17
19
RAM
ENGINE I/F
PRINTER ENGINE
16
FONT ROM
13
PROGRAM ROM
DATA ROM
OPERATION UNIT
1012
20
MC
EXT MEM
14
CLIENT COMPUTER
5000

FIG. 3

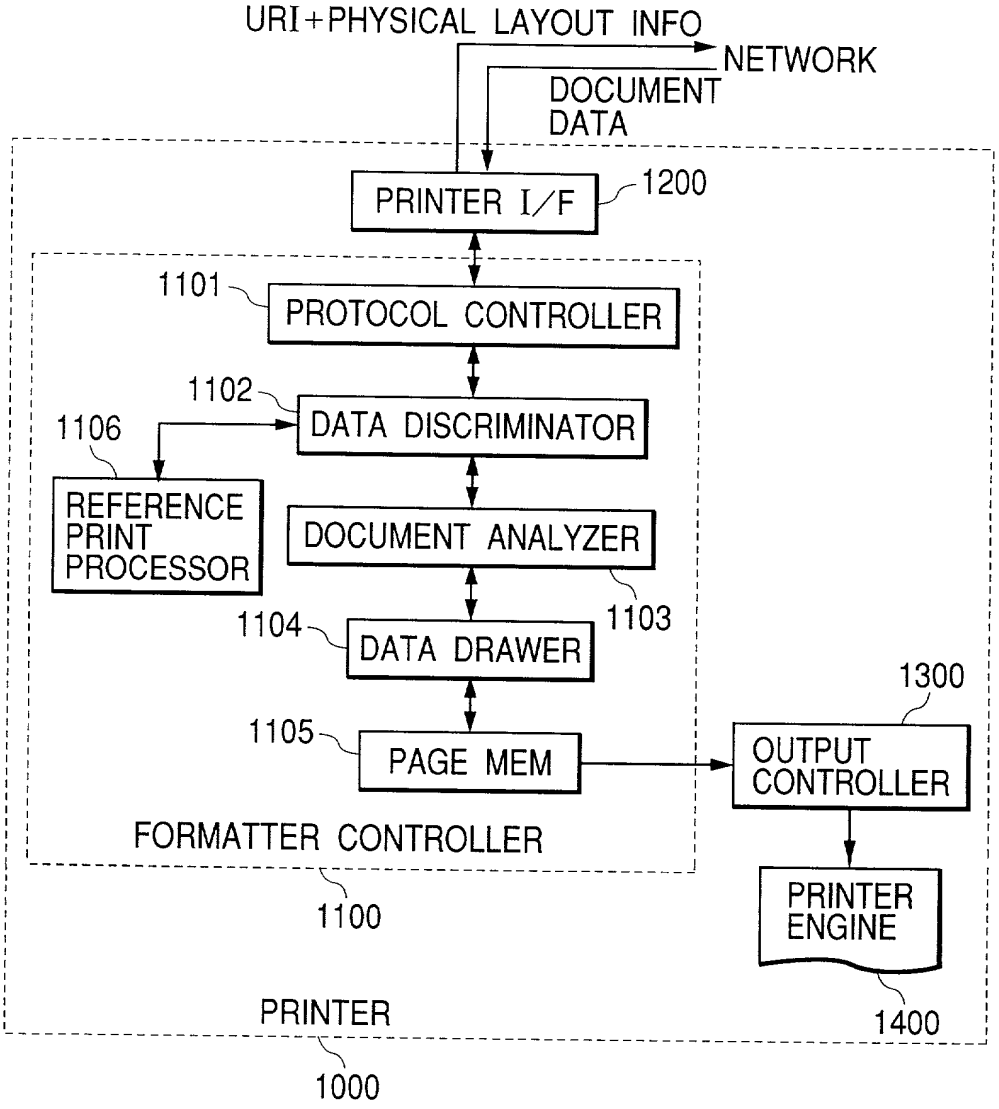


FIG. 4

```
<order reference-print>
<document="http://myserver.com/mydocument"/>
<papersize>A4</paper>
<orientation>portrait</orientation>
</order>
```

FIG. 5

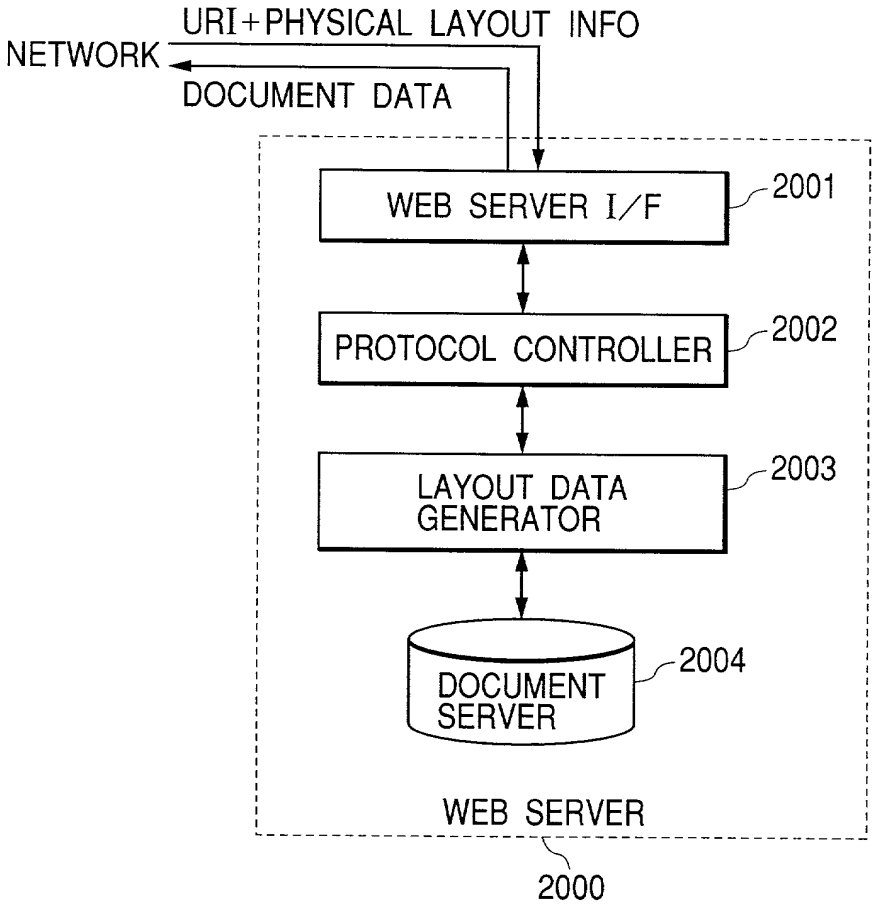


FIG. 6

```
<xml stylesheet="\mystylesheet">
<doc>
<title>Sample</title>
<para>
This document is written in Markup
Language, logical data structure and lo
gical layout.
</para>
<table></table>
<tfooter>table 1</tfooter>
</doc>
```

FIG. 7

```
<stylesheet>
<template pattern="title">
<font size=big fontcolor=red position=center>
</template>

<template pattern="para">
<fontsize=small fontcolor=black>
<pat=mesh patcolor=blue>
</template>

<template pattern="table">
<table width=4 height=3>
</template>

<template pattern="tfooter">
<font size=middle position=center>
</template>
</stylesheet>
```

FIG. 8

```
<document>
<unit size=mm/>
<text size=24po color=red x=100 y=0>
Sample</text>
<fill pat="0xaa aa"color=blue></fill>
<rect 10 200 1000 300/>
<text size=10po color=black x=0 y=30>
This document is written in</text>
<text x=20 y=30>
Markup Language, logical data</text>
<text x=40 y=30>
structure and logical layout.</text>
<fill pat=null/>
<rect 40 50 200 100/>
<line 80 50 80 100/>
<line 120 50 80 100/>
<line 160 50 80 100/>
<line 200 50 80 100/>
<line 40 70 200 70/>
<line 40 90 200 90/>
</document>
```

0597529 - 11/15/96

FIG. 9

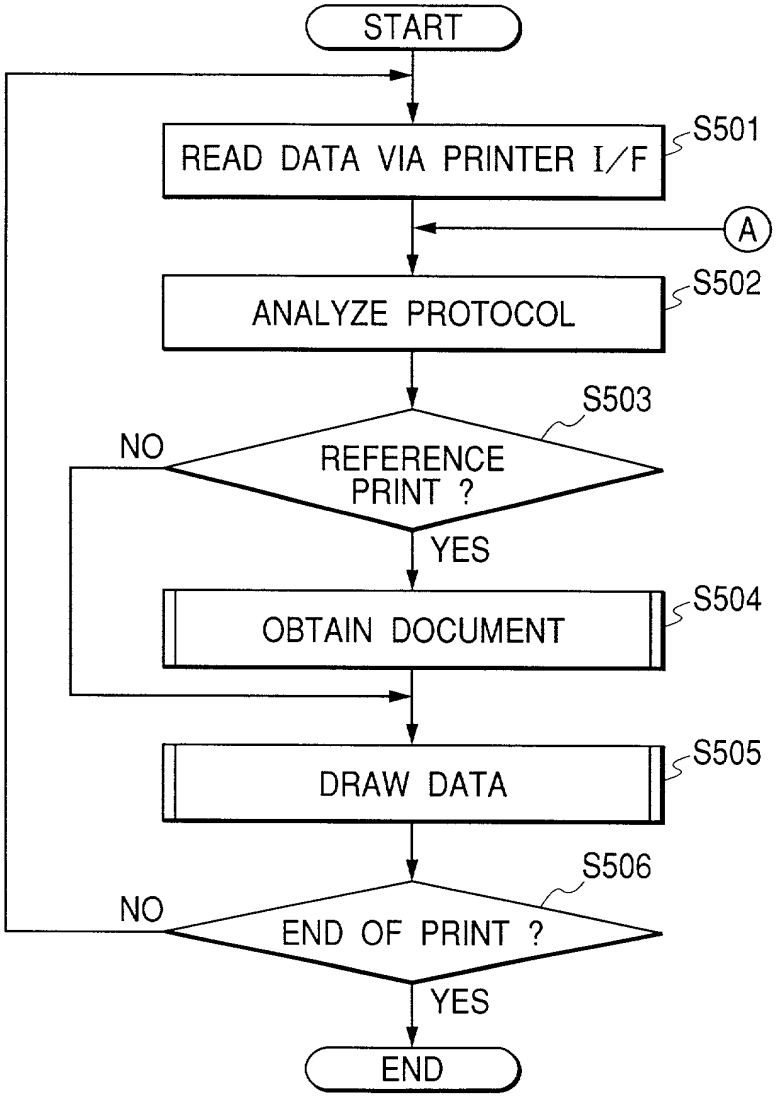

START
READ DATA VIA PRINTER I/F
S501
A
ANALYZE PROTOCOL
S502
S503
NO
REFERENCE PRINT ?
YES
OBTAIN DOCUMENT
S504
DRAW DATA
S505
S506
NO
END OF PRINT ?
YES
END

FIG. 10

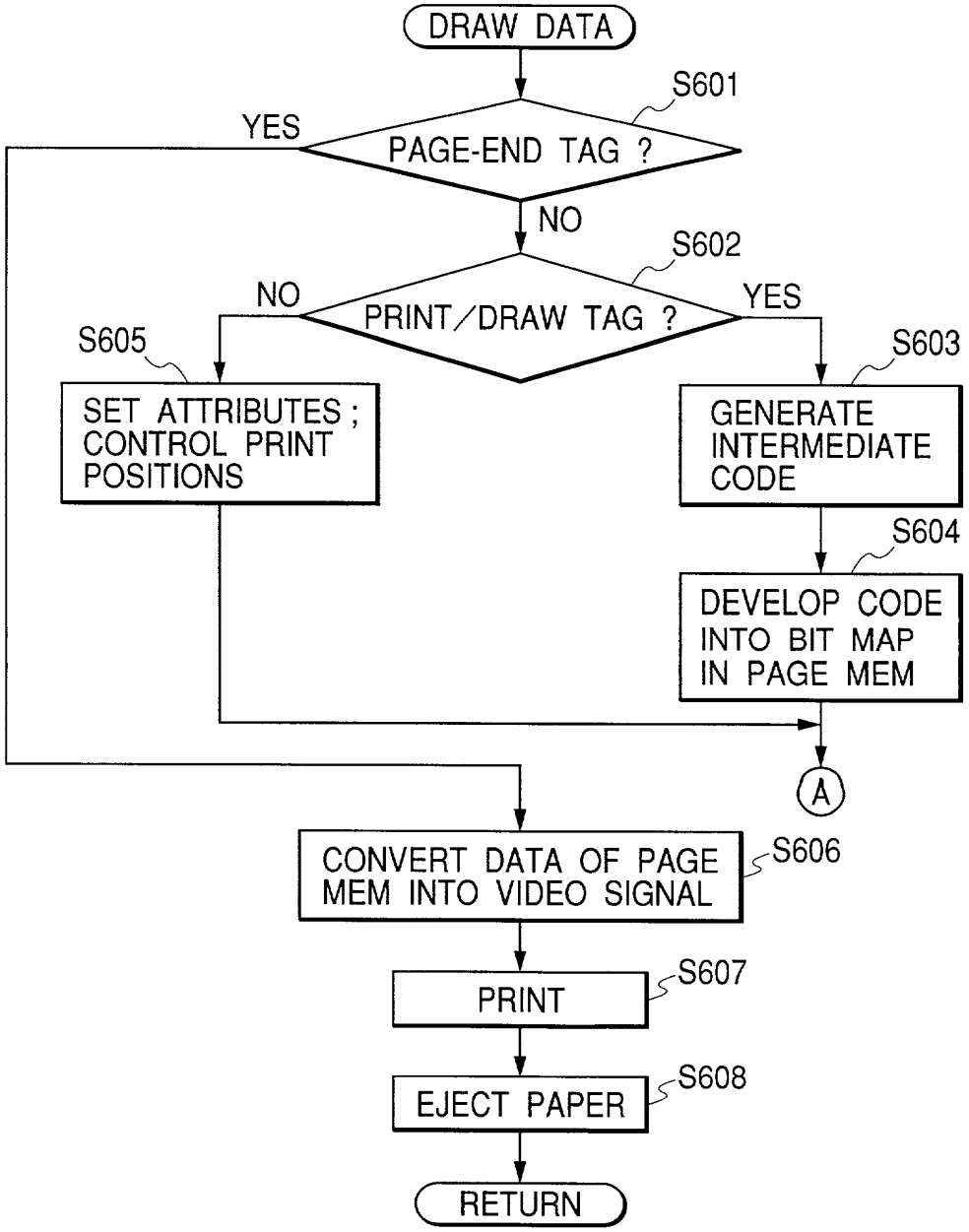

DRAW DATA
S601
PAGE-END TAG ?
YES
NO
S602
PRINT/DRAW TAG ?
NO
YES
S605
SET ATTRIBUTES ; CONTROL PRINT POSITIONS
S603
GENERATE INTERMEDIATE CODE
S604
DEVELOP CODE INTO BIT MAP IN PAGE MEM
A
CONVERT DATA OF PAGE MEM INTO VIDEO SIGNAL
S606
PRINT
S607
EJECT PAPER
S608
RETURN

FIG. 11

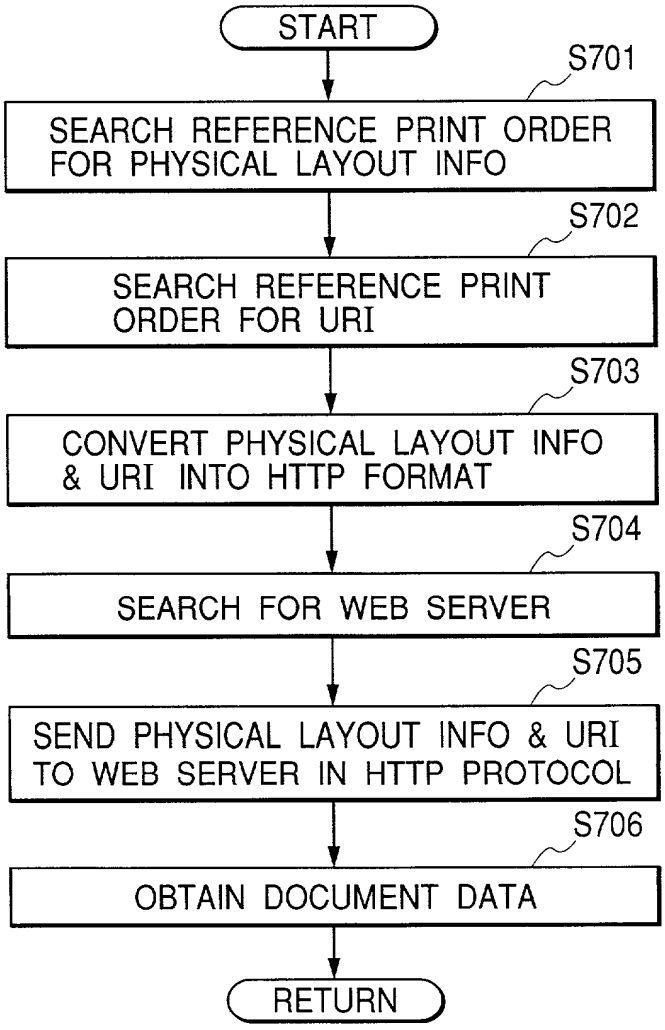
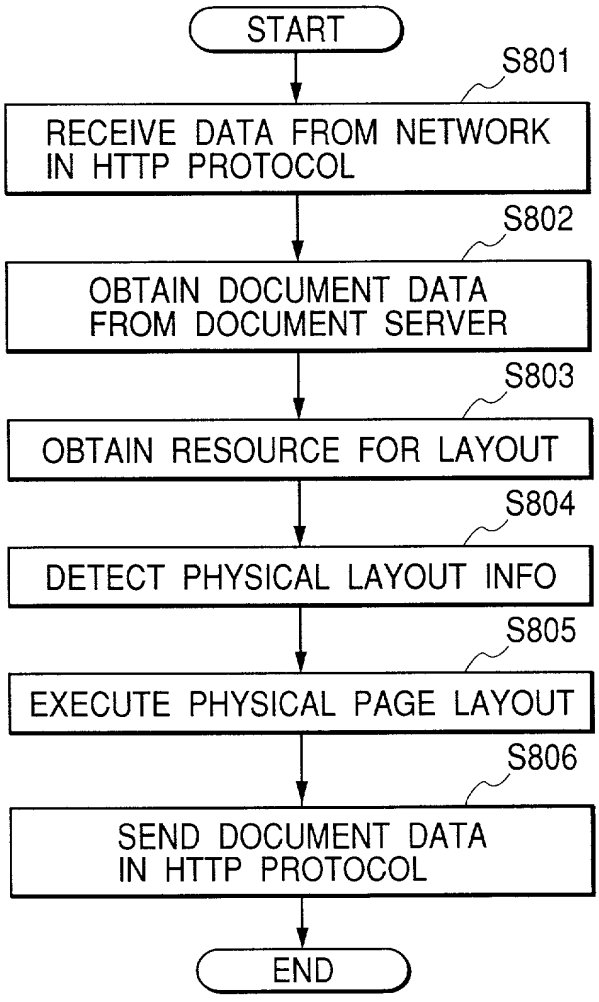

START
S701
SEARCH REFERENCE PRINT ORDER FOR PHYSICAL LAYOUT INFO
S702
SEARCH REFERENCE PRINT ORDER FOR URI
S703
CONVERT PHYSICAL LAYOUT INFO & URI INTO HTTP FORMAT
S704
SEARCH FOR WEB SERVER
S705
SEND PHYSICAL LAYOUT INFO & URI TO WEB SERVER IN HTTP PROTOCOL
S706
OBTAIN DOCUMENT DATA
RETURN

FIG. 12



EPSON-EP-0952536

FIG. 13

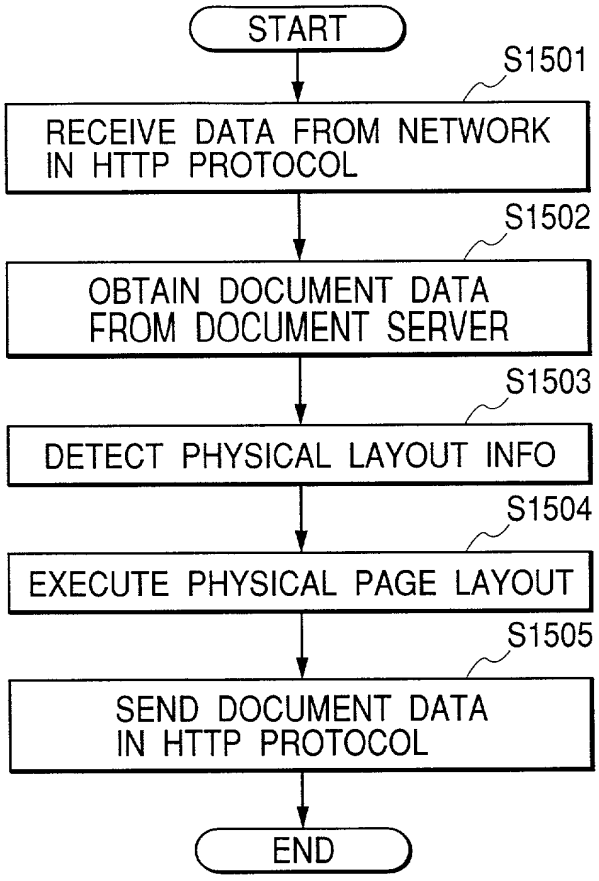


FIG. 14

```
<html>
<body bgcolor="#ffffff">
<H1>Sample</H1>
<table border="0"width="100%">
<tr>
  <td width="60%"><p align="center">Sample</td>
  <td width="20%"><p align="right">Data</td>
 </tr>
</table>
<hr>
<p><font size="4">html document</font></p>
</body>
</html>
```

FIG. 15

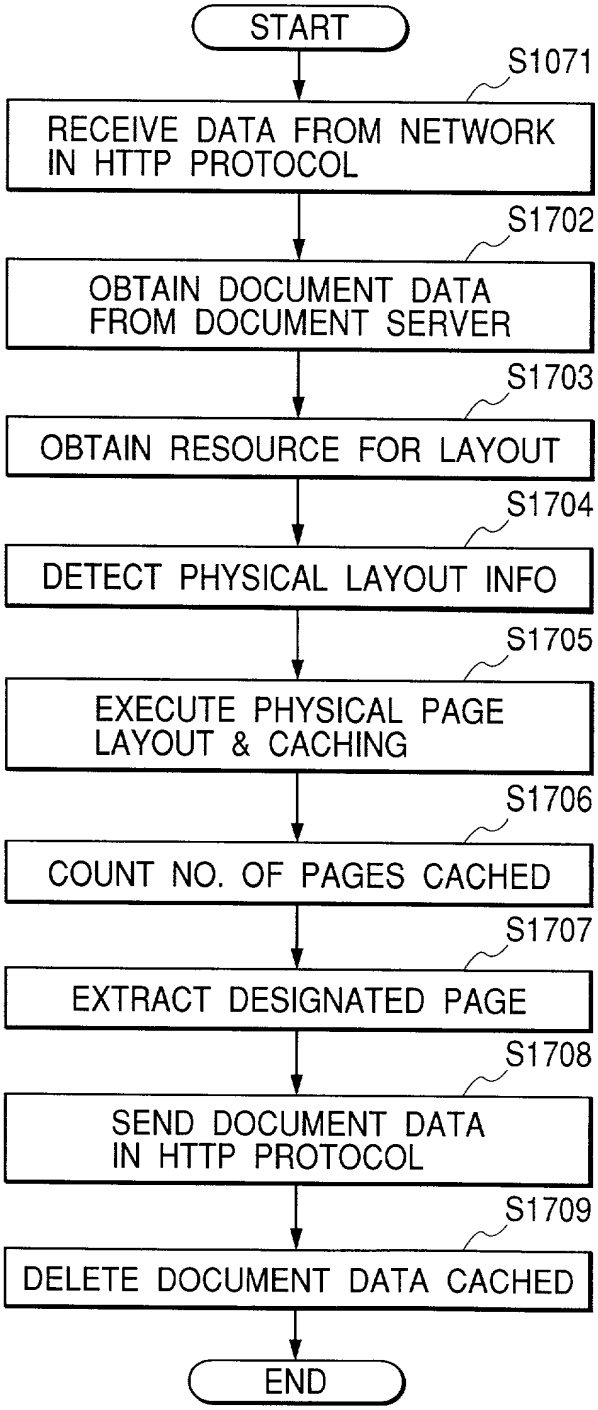

START
S1071
RECEIVE DATA FROM NETWORK IN HTTP PROTOCOL
S1702
OBTAIN DOCUMENT DATA FROM DOCUMENT SERVER
S1703
OBTAIN RESOURCE FOR LAYOUT
S1704
DETECT PHYSICAL LAYOUT INFO
S1705
EXECUTE PHYSICAL PAGE LAYOUT & CACHING
S1706
COUNT NO. OF PAGES CACHED
S1707
EXTRACT DESIGNATED PAGE
S1708
SEND DOCUMENT DATA IN HTTP PROTOCOL
S1709
DELETE DOCUMENT DATA CACHED
END

FIG. 16

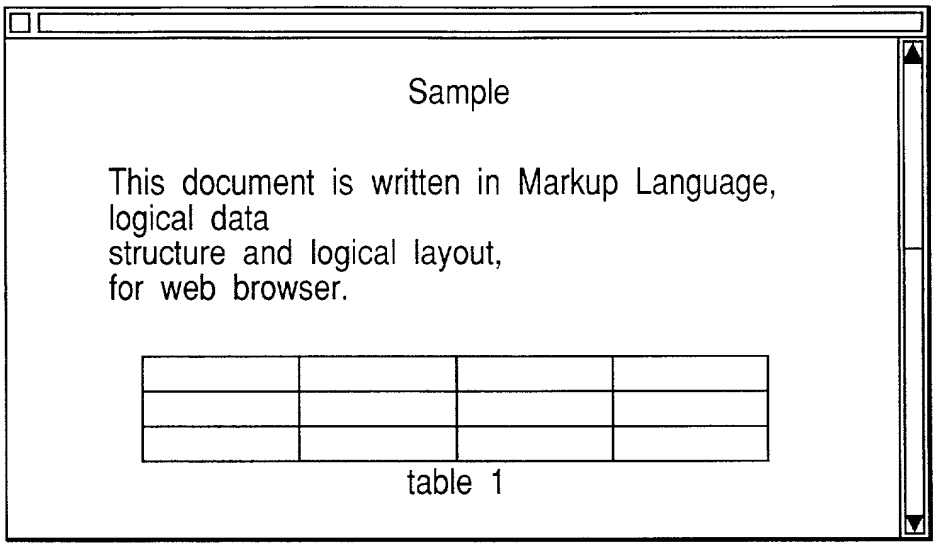
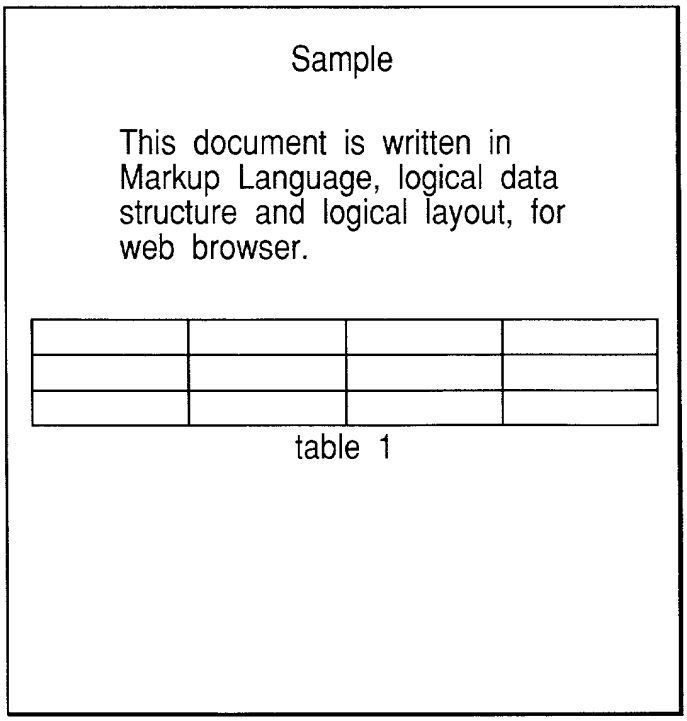


FIG. 17



EP 0 952 528 A1